

IN THE CLAIMS:

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A method for configuring a first node in a graphical program, the method comprising:

displaying ~~[[a]]~~ the first node in ~~[[a]]~~ the graphical program, wherein the first node is configurable to perform a plurality of operations depending upon user input specifying configuration information for the first node, wherein the graphical program comprises a first plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

displaying a graphical user interface (GUI) for specifying configuration information for the first node, wherein the GUI ~~[[comprises]]~~ displays information indicating the plurality of operations which the first node is configurable to perform, wherein the GUI is useable [[in configuring]] to configure the first node to perform one or more operations from the plurality of operations;

receiving user input ~~[[via]]~~ to the GUI specifying one or more ~~[[desired]]~~ operations for the first node to perform from the plurality of operations; and

configuring the first node to perform the one or more specified operations in response to the user input to the GUI, wherein configuring the first node comprises automatically [[programmatically]] generating graphical source code for the first node to implement the one or more [[desired]] specified operations, in response to the user input wherein automatically generating the graphical source code for the first node comprises automatically generating a second plurality of interconnected nodes without receiving user input directly specifying the nodes in the second plurality of interconnected nodes, wherein the second plurality of interconnected nodes is operable to perform the one or more specified operations during execution of the graphical program.

2. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical source code for the first node to implement the one or more [[desired]] specified operations does not include generating graphical source code corresponding to operations from the plurality of operations that are not among the one or more [[desired]] specified operations.

3. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical source code for the first node to implement the one or more [[desired]] specified operations does not include generating graphical source code not necessary to implement the one or more [[desired]] specified operations.

4. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical source code for the first node to implement the one or more [[desired]] specified operations comprises generating a minimal amount of graphical source code to implement the one or more [[desired]] specified operations.

5. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical source code for the first node comprises [[programmatically]] automatically generating the graphical source code as a sub-program of the graphical program, wherein the sub-program encapsulates the second plurality of interconnected nodes, wherein the first node represents the sub-program.

6. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical source code for the first node comprises automatically replacing the first node in the graphical program with the second plurality of interconnected nodes, wherein replacing the first node with the second plurality of interconnected nodes comprises displaying the second

plurality of interconnected nodes in the graphical program in place of the first node programmatically generated graphical source code.

7. (Currently Amended) The method of claim 1, wherein the one or more ~~[[desired]]~~ specified operations is a first one or more ~~[[desired]]~~ operations, the method further comprising:

~~receiving user input requesting to change configuration information for the node; after said programmatically generating the graphical source code for the node;~~

~~re-displaying the graphical user interface (GUI) in response to the user input requesting to change the configuration information of the node;~~

receiving user input ~~[[via]]~~ to the GUI specifying a second one or more ~~[[desired]]~~ operations for the first node to perform from the plurality of operations, after said automatically generating the graphical source code for the first node; and

~~[[programmatically]]~~ automatically replacing the previously generated graphical source code with new automatically generated graphical source code for the first node, wherein the new graphical source code implements the second one or more ~~[[desired]]~~ operations.

8. (Currently Amended) The method of claim 7,

wherein the first one or more ~~[[desired]]~~ operations includes a first operation, wherein the previously generated graphical source code for the first node includes graphical source code to implement the first operation;

wherein the second one or more ~~[[desired]]~~ operations does not include the first operation;

wherein the new graphical source code automatically generated for the first node does not include graphical source code to implement the first operation.

9. (Currently Amended) The method of claim 1,

wherein ~~no functionality is set for the first node is not operable to perform any function in the graphical program~~ until after said ~~[[programmatically]]~~ automatically

generating the graphical source code for the first node to implement the one or more specified operations.

10. (Currently Amended) The method of claim 1,
wherein default functionality is set for the first node before said automatically generating the graphical source code for the first node;

wherein said [[programmatically]] automatically generating the graphical source code for the first node comprises replacing the default functionality for the first node with functionality implemented by the [[programmatically]] automatically generated graphical source code.

11. (Currently Amended) The method of claim 1,
wherein no program instructions to be executed during execution of the graphical program are associated with the first node until after said [[programmatically]] automatically generating the graphical source code for the first node.

12. (Currently Amended) The method of claim 1, further comprising:
receiving user input requesting to specify configuration information for the first node;

wherein said displaying the graphical user interface (GUI) is performed in response to the user input requesting to specify configuration information for the first node.

13. (Currently Amended) The method of claim 1,
wherein the GUI for specifying configuration information for the first node comprises one or more GUI input panels, wherein the one or more GUI input panels include GUI input controls;

wherein said receiving user input to the GUI specifying the one or more operations for the first node to perform comprises receiving user input to the GUI controls in the one or more GUI input panels specifying the one or more operations for the first node to perform.

~~wherein the one or more GUI input panels include GUI input controls operable to receive user input for configuring functionality for the node.~~

14. (Canceled)

15. (Currently Amended) A method for configuring a first node in a graphical program, the method comprising:

displaying ~~[[a]]~~ the first node in ~~[[a]]~~ the graphical program, wherein the first node is configurable to perform functionality depending upon user input specifying configuration information for the first node, ~~[[and]]~~ wherein the graphical program comprises a first plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

displaying a graphical user interface (GUI) for specifying configuration information for the first node, wherein the GUI displays information indicating possible functionality which the first node can be configured to perform ~~is useable to specify functionality for the node;~~

receiving user input ~~[[via]]~~ to the GUI ~~[[specifying]]~~ selecting desired functionality for the first node from the indicated possible functionality; and

configuring the first node to perform the desired functionality selected from the GUI in response to the user input selecting the desired functionality, wherein configuring the first node comprises automatically [[programmatically]] generating graphical source code for the first node to implement the [[specified]] desired functionality selected from the GUI, in response to the user input wherein automatically generating the graphical source code for the first node comprises automatically generating a second plurality of interconnected nodes without receiving user input directly specifying the nodes in the second plurality of interconnected nodes, wherein the second plurality of interconnected nodes is operable to perform the desired functionality selected from the GUI during execution of the graphical program.

16. (Currently Amended) The method of claim 15,

wherein the GUI is useable to ~~[[specify]]~~ select first functionality and second functionality for the first node, wherein the GUI displays information indicating that the first node can be configured to perform the first functionality and the second functionality;

wherein the user input ~~[[specifying]]~~ selecting the desired functionality ~~[[specifies]]~~ for the first node selects the first functionality but does not ~~[[specify]]~~ select the second functionality;

wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node includes ~~[[programmatically]]~~ automatically generating graphical source code to implement the first functionality;

wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node does not include ~~[[programmatically]]~~ automatically generating graphical source code to implement the second functionality.

17. (Currently Amended) A computer-readable memory medium comprising program instructions for configuring a first node in a graphical program, wherein the program instructions are executable to:

display ~~[[a]]~~ the first node in ~~[[a]]~~ the graphical program, wherein the first node is configurable to perform a plurality of operations depending upon user input specifying configuration information for the first node, wherein the graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

display a graphical user interface (GUI) for specifying configuration information for the first node, wherein the GUI displays information indicating the plurality of operations which the first node is configurable to perform, wherein the GUI ~~comprises information is~~ useable in ~~guiding a user in configuring to configure~~ the first node to perform one or more operations from the plurality of operations;

receive user input ~~[[via]]~~ to the GUI specifying one or more ~~[[desired]]~~ operations for the first node to perform from the plurality of operations; and

configure the first node to perform the one or more specified operations in response to the user input to the GUI, wherein configuring the first node comprises

automatically generating ~~programmatically generate~~ graphical source code for the first node to implement the one or more [[desired]] specified operations, in response to the user input wherein automatically generating the graphical source code for the first node comprises automatically generating a second plurality of interconnected nodes without receiving user input directly specifying the nodes in the second plurality of interconnected nodes, wherein the second plurality of interconnected nodes is operable to perform the one or more specified operations during execution of the graphical program.

18. (Currently Amended) The memory medium of claim 17,
wherein said [[programmatically]] automatically generating the graphical source code for the first node to implement the one or more [[desired]] specified operations does not include generating graphical source code corresponding to operations from the plurality of operations that are not among the one or more [[desired]] specified operations.

19. (Currently Amended) The memory medium of claim 17,
wherein said [[programmatically]] automatically generating the graphical source code for the first node to implement the one or more [[desired]] specified operations does not include generating graphical source code not necessary to implement the one or more [[desired]] specified operations.

20. (Currently Amended) The memory medium of claim 17,
wherein said [[programmatically]] automatically generating the graphical source code for the first node to implement the one or more [[desired]] specified operations comprises generating a minimal amount of graphical source code to implement the one or more [[desired]] specified operations.

21. (Currently Amended) The memory medium of claim 17,
wherein said [[programmatically]] automatically generating the graphical source code for the first node comprises [[programmatically]] automatically generating the graphical source code as a sub-program of the graphical program, wherein the sub-

program encapsulates the second plurality of interconnected nodes, wherein the first node represents the sub-program.

22. (Currently Amended) The memory medium of claim 17,
wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node comprises automatically replacing the first node in the graphical program with the second plurality of interconnected nodes, wherein replacing the first node with the second plurality of interconnected nodes comprises displaying the second plurality of interconnected nodes in the graphical program in place of the first node ~~programmatically generated graphical source code~~.

23. (Currently Amended) The memory medium of claim 17,
wherein ~~no functionality is set for the first node~~ is not operable to perform any function in the graphical program until after said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node to implement the one or more specified operations.

24. (Currently Amended) The memory medium of claim 17,
wherein default functionality is set for the first node before said automatically generating the graphical source code for the first node;
wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node comprises replacing the default functionality for the first node with functionality implemented by the ~~[[programmatically]]~~ automatically generated graphical source code.

25. (Currently Amended) The memory medium of claim 17,
wherein no program instructions to be executed during execution of the graphical program are associated with the first node until after said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node.

26. (Currently Amended) A computer-readable memory medium comprising program instructions for configuring a first node in a graphical program, wherein the program instructions are executable to:

display ~~[[a]]~~ the first node in ~~[[a]]~~ the graphical program, wherein the first node is configurable to perform functionality depending upon user input specifying configuration information for the first node, wherein the graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

display a graphical user interface (GUI) for specifying configuration information for the first node, wherein the GUI displays information indicating possible functionality which the first node can be configured to perform ~~is useable to specify functionality for the node;~~

receive user input ~~[[via]]~~ to the GUI ~~[[specifying]]~~ selecting desired functionality for the first node from the indicated possible functionality; and

configure the first node to perform the desired functionality selected from the GUI in response to the user input selecting the desired functionality, wherein configuring the first node comprises automatically generating ~~programmatically generate~~ graphical source code for the first node to implement the ~~[[specified]]~~ desired functionality selected from the GUI, ~~in response to the user input~~ wherein automatically generating the graphical source code for the first node comprises automatically generating a second plurality of interconnected nodes without receiving user input directly specifying the nodes in the second plurality of interconnected nodes, wherein the second plurality of interconnected nodes is operable to perform the desired functionality selected from the GUI during execution of the graphical program.

27. (Currently Amended) The memory medium of claim 26,

wherein the GUI is useable to ~~[[specify]]~~ select first functionality and second functionality for the first node, wherein the GUI displays information indicating that the first node can be configured to perform the first functionality and the second functionality;

wherein the user input ~~[[specifying]]~~ selecting the desired functionality ~~[[specifies]]~~ for the first node selects the first functionality but does not ~~[[specify]]~~ select the second functionality;

wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node includes ~~[[programmatically]]~~ automatically generating graphical source code to implement the first functionality;

wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node does not include ~~[[programmatically]]~~ automatically generating graphical source code to implement the second functionality.

28. (Currently Amended) The method of claim 1, further comprising:

displaying the ~~[[programmatically]]~~ automatically generated graphical source code in the graphical program, wherein displaying the automatically generated graphical source code in the graphical program comprises displaying the second plurality of interconnected nodes in the graphical program.

29. (Currently Amended) The method of claim 1,

wherein said ~~[[programmatically]]~~ automatically generating the graphical source code for the first node comprises automatically displaying the ~~[[programmatically]]~~ automatically generated graphical source code in place of the first node in the graphical program.

30. (Currently Amended) The method of claim 1, further comprising:

receiving user input selecting the first node prior to said displaying the first node in the graphical program.

31. (Currently Amended) The method of claim 1,

wherein the ~~programmatically generated graphical source code comprises a~~ second plurality of interconnected nodes ~~[[that]]~~ visually ~~[[indicate]]~~ indicates functionality of the one or more specified operations ~~graphical source code.~~

32. (Currently Amended) The method of claim 1,
wherein the ~~programmatically generated graphical source code comprises a~~ nodes
in the second plurality of interconnected nodes are interconnected in one or more of a data
flow, control flow, and/or execution flow format.

33. (Currently Amended) The method of claim 1,
wherein the ~~programmatically generated graphical source code comprises a~~ nodes
in the second plurality of interconnected nodes are interconnected to indicate data flow
among the nodes in the second plurality of interconnected nodes.

34. (Currently Amended) The method of claim 1,
wherein the ~~programmatically generated graphical source code comprises a~~ nodes
in the second plurality of interconnected nodes are interconnected to indicate control flow
among the nodes in the second plurality of interconnected nodes.

35. (Currently Amended) The method of claim 1,
wherein the ~~programmatically generated graphical source code comprises a~~ nodes
in the second plurality of interconnected nodes are interconnected to indicate execution
flow among the nodes in the second plurality of interconnected nodes.

36. (Currently Amended) The memory medium of claim 26, wherein the program
instructions are further executable to display the ~~[[programmatically]]~~ automatically
generated graphical source code in the graphical program, wherein displaying the
automatically generated graphical source code in the graphical program comprises
displaying the second plurality of interconnected nodes in the graphical program.

37. (Currently Amended) The memory medium of claim 26,
wherein, in ~~[[programmatically]]~~ automatically generating the graphical source
code for the first node, the program instructions are executable to ~~[[programmatically]]~~
automatically generate the graphical source code as a sub-program of the graphical

program, wherein the sub-program encapsulates the second plurality of interconnected nodes, wherein the first node represents the sub-program.

38. (Currently Amended) The memory medium of claim 26,
wherein, in ~~[[programmatically]]~~ automatically generating the graphical source code for the first node, the program instructions are executable to automatically display the ~~[[programmatically]]~~ automatically generated graphical source code in place of the first node in the graphical program.

39. (Currently Amended) The memory medium of claim 26, wherein the program instructions are further executable to receive user input selecting the first node prior to said displaying the first node in the graphical program.

40. (Currently Amended) The memory medium of claim 26,
wherein the ~~programmatically generated graphical source code comprises a~~ second plurality of interconnected nodes implements and ~~[[that]] visually [[indicate]]~~ indicates the desired functionality selected from ~~[[of]] the GUI graphical source code.~~

41. (Currently Amended) The memory medium of claim 26,
wherein the ~~programmatically generated graphical source code comprises a~~ nodes in the second plurality of interconnected nodes are interconnected in one or more of a data flow, control flow, and/or execution flow format.

42. (Currently Amended) The memory medium of claim 26,
wherein the ~~programmatically generated graphical source code comprises a~~ nodes in the second plurality of interconnected nodes are interconnected to indicate data flow among the nodes in the second plurality of interconnected nodes.

43. (Currently Amended) The memory medium of claim 26,
wherein the ~~[[specified]]~~ desired functionality selected from the GUI is first desired functionality;

wherein the program instructions are further executable to:

~~receive user input requesting to change configuration information for the node, after said programmatically generating the graphical source code for the node;~~

~~re-display the graphical user interface (GUI) in response to the user input requesting to change configuration information for the node;~~

receive user input [[via]] to the GUI [[specifying]] selecting second desired functionality for the first node from the indicated possible functionality; and

[[programmatically]] automatically generate new graphical source code for the first node in place of the previously generated graphical source code, wherein the new graphical source code implements the second desired functionality.

44. (Previously Presented) The memory medium of claim 43,

wherein the program instructions are further executable to display the new graphical source code in place of the previously generated graphical source code.

45. (Currently Amended) The memory medium of claim 26,

wherein ~~no functionality is set for the first node is not operable to perform any function in the graphical program~~ until after the graphical source code is [[programmatically]] automatically generated for the first node to implement the one or more specified operations.

46. (Currently Amended) The memory medium of claim 26,

wherein default functionality is set for the first node before said automatically generating the graphical source code for the first node;

wherein said [[programmatically]] automatically generating the graphical source code for the first node comprises replacing the default functionality for the first node with the functionality implemented by the [[programmatically]] automatically generated graphical source code.

47. (Currently Amended) The memory medium of claim 26,

wherein no program instructions to be executed during execution of the graphical program are associated with the first node until after the graphical source code is [[programmatically]] automatically generated for the first node.

48. (Currently Amended) A method for creating a graphical program, the method comprising:

selecting a first graphical program node for inclusion in [[a]] the graphical program in response to user input, wherein the graphical program comprises a first plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

displaying a graphical user interface (GUI) for configuring operation of the first graphical program node;

receiving user input to the GUI configuring desired operation of the first graphical program node;

[[programmatically]] automatically generating graphical source code for the first graphical program node based on the user input configuring desired operation of the first graphical program node, wherein automatically generating the graphical source code for the first graphical program node comprises automatically generating a second plurality of interconnected nodes without receiving user input directly specifying the nodes in the second plurality of interconnected nodes; and

displaying the [[programmatically]] automatically generated graphical source code, wherein displaying the automatically generated graphical source code comprises displaying the second plurality of interconnected nodes.

49. (Currently Amended) The method of claim 48,

wherein said displaying the [[programmatically]] automatically generated graphical source code comprises displaying the [[programmatically]] automatically generated graphical source code in place of the first graphical program node in the graphical program.

50. (Currently Amended) The method of claim 48,

wherein said receiving user input to the GUI configuring desired operation of the first graphical program node comprises receiving user input to the GUI configuring first operation of the first graphical program node;

wherein the method further comprises:

~~receiving user input requesting to change operation of the node, after said programmatically generating the graphical source code;~~

~~re-displaying the graphical user interface (GUI) in response to the user input requesting to change operation of the node;~~

receiving user input to the GUI configuring second operation of the first graphical program node; and

[[programmatically]] automatically replacing the previously generated graphical source code with new automatically generated graphical source code, based on the user input configuring ~~wherein the new graphical source code implements~~ the second operation.

51. (Currently Amended) The method of claim 48, further comprising:

displaying the first graphical program node in the graphical program in response to said selecting the first graphical program node.

52. (Currently Amended) The method of claim 48,

wherein the ~~programmatically generated graphical source code comprises a~~ automatically generated second plurality of interconnected nodes implements and [[that]] visually [[indicate]] indicates the desired operation configured by the user input to the GUI functionality of the graphical source code.

53. (Currently Amended) The method of claim 48,

wherein the ~~programmatically generated graphical source code comprises a~~ nodes in the second plurality of interconnected nodes are interconnected in one or more of a data flow, control flow, and/or execution flow format.

54. (Currently Amended) The method of claim 48,

wherein the ~~programmatically generated graphical source code comprises a~~ nodes in the second plurality of interconnected nodes ~~are~~ interconnected to indicate data flow among the nodes in the second plurality of interconnected nodes.

55. (Currently Amended) The method of claim 48,
wherein said ~~[[programmatically]]~~ automatically generating the graphical source code comprises ~~[[programmatically]]~~ automatically generating the graphical source code as a sub-program ~~[[of]]~~ encapsulated by the first node, wherein the sub-program includes the second plurality of interconnected nodes.

56. (Previously Presented) The method of claim 48,
wherein the GUI comprises at least one panel.

57. (Previously Presented) The method of claim 48,
wherein the GUI comprises a plurality of panels.

58. (Currently Amended) A method for creating a graphical program, the method comprising:

selecting a first graphical program node in response to user input;

displaying the first graphical program node in ~~[[a]]~~ the graphical program after said selecting, wherein the graphical program comprises a first plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

displaying a graphical user interface (GUI) after selecting the first graphical program node;

receiving user input to the GUI configuring desired operation of the first graphical program node; and

~~[[programmatically]]~~ automatically generating graphical source code based on the user input configuring desired operation of the first graphical program node, wherein automatically generating the graphical source code ~~[[is programmatically]]~~ comprises

automatically including a second plurality of interconnected nodes in [[generated as]] a sub-program [[of]] encapsulated by the first graphical program node.

59. (Currently Amended) The method of claim 58, further comprising:

receiving user input selecting the first node after said [[programmatically]] automatically generating the graphical source code; and

displaying the [[programmatically]] automatically generated graphical source code in response to the user input selecting the first node, wherein displaying the automatically generated graphical source code comprises displaying the second plurality of interconnected nodes.

60. (Currently Amended) A computer-readable memory medium comprising program instructions for configuring a first graphical program node in a graphical program, wherein the program instructions are executable to implement:

displaying [[a]] the first graphical program node in [[a]] the graphical program in response to user input, wherein the graphical program comprises a first plurality of interconnected nodes that visually indicate functionality of the graphical program, and wherein the graphical program is executable to perform the functionality;

displaying a graphical user interface (GUI) after displaying the first graphical program node;

receiving user input to the GUI configuring the first graphical program node;

[[programmatically]] automatically generating graphical source code based on the user input configuring the first graphical program node, wherein the graphical source code is automatically generated without user input directly specifying the graphical source code; and

displaying the [[programmatically]] automatically generated graphical source code.

61. (Currently Amended) The [[method]] memory medium of claim 60,

wherein said displaying the [[programmatically]] automatically generated graphical source code comprises displaying the [[programmatically]] automatically generated graphical source code in place of the first node in the graphical program.

62. (Currently Amended) The memory medium of claim 60,
wherein said displaying the [[programmatically]] automatically generated graphical source code comprises displaying the [[programmatically]] automatically generated graphical source code in the graphical program;

wherein the first graphical program node is no longer displayed in the graphical program at least as of when the [[programmatically]] automatically generated graphical source code is displayed in the graphical program.

63. (Currently Amended) The memory medium of claim 60,
wherein said displaying the [[programmatically]] automatically generated graphical source code comprises displaying the [[programmatically]] automatically generated graphical source code in the graphical program;

wherein the program instructions are further executable to implement:

discontinuing displaying the first graphical program node in the graphical program after displaying the [[programmatically]] automatically generated graphical source code in the graphical program.

64. (Currently Amended) The memory medium of claim 60,
wherein the [[programmatically]] automatically generated graphical source code comprises a plurality of interconnected nodes that visually indicate functionality of the graphical source code.

65. (Currently Amended) The memory medium of claim 60,
wherein the [[programmatically]] automatically generated graphical source code comprises a plurality of nodes interconnected in one or more of a data flow, control flow, and/or execution flow format.

66. (New) The method of claim 1, further comprising:

executing the graphical program, wherein executing the graphical program comprises executing the first node, wherein executing the first node comprises executing the automatically generated graphical source code, wherein executing the automatically generated graphical source code comprises executing the second plurality of interconnected nodes to perform the one or more specified operations which the first node is configured to perform.

67. (New) The method of claim 1,

wherein said automatically generating the second plurality of interconnected nodes comprises automatically generating the second plurality of interconnected nodes as a sub-program encapsulated by the first node.

68. (New) The method of claim 1,

wherein said automatically generating the second plurality of interconnected nodes comprises automatically generating a plurality of connections among the nodes without user input directly specifying the connections.

69. (New) The method of claim 1,

wherein said automatically generating the second plurality of interconnected nodes comprises automatically including the second plurality of interconnected nodes in the graphical program without user input directly requesting the second plurality of interconnected nodes to be included in the graphical program.

70. (New) The method of claim 1, further comprising:

automatically displaying the second plurality of interconnected nodes without user input directly requesting the second plurality of interconnected nodes to be displayed.